

Family Name: Other Names:

Student ID: Signature

COMP 102: Test

2018, May 14 ** WITH SOLUTIONS **

Instructions

- Time allowed: **50 minutes**
- Attempt **all** the questions. There are 50 marks in total.
- Write your answers in this test paper and hand in all sheets.
- If you think some question is unclear, ask for clarification.
- Brief Java documentation is provided with the test
- This test contributes 15% of your final grade
(But your mark will be increased to your exam mark if that is higher.)
- You may use dictionaries.
- You may write notes and working on this paper, but make sure your answers are clear.

Questions

Marks

1. Understanding Parameters	[6]	<input type="text"/>
2. Writing with while	[5]	<input type="text"/>
3. Defining Classes	[15]	<input type="text"/>
4. Event Driven Input	[12]	<input type="text"/>
5. Files	[12]	<input type="text"/>
	TOTAL:	<input type="text"/>

SPARE PAGE FOR EXTRA ANSWERS

Cross out rough working that you do not want marked.
Specify the question number for work that you do want marked.

Question 1. Understanding Parameters**[6 marks]**

What will the following printOut method print out?.

```
public void printOut() {
    String name1 = "Alex";
    String name2 = "Bob";
    String note1 = "call back";
    String note2 = "coffee";

    this.printStuff(name1, note1);
    this.printStuff(name1, name2);
    this.printStuff(note2, name1);
}

public void printStuff (String name, String note){
    UI.println (name + ": " + note);
}
```

```
Alex: call back
Alex: Bob
coffee: Alex
```

SPARE PAGE FOR EXTRA ANSWERS

Cross out rough working that you do not want marked.
Specify the question number for work that you do want marked.

Question 2. Writing with while**[5 marks]**

Complete the showTable method to display the following table that converts from Celsius (0, 2, 4, 6,...,100) to Fahrenheit. Note that Fahrenheit = Celsius * 9/5 +32.

Celsius	Fahrenheit
0	32.0
2	35.6
4	39.2
6	42.8
...	
98	208.4
100	212.0

```
public void showTable() {  
    Ul.println ("Celsius Farenheit");  
    double c = 0;  
    while(c<=100) {  
        double f = c * 9/5 + 32;  
        Ul.printf ( "%.0f\t%.1f\n", c, f);  
        c = c + 2;  
    }  
}
```

Question 3. Defining Classes**[15 marks]**

- (a) [10 marks] Design a class named Rectangle to represent a rectangle. The class contains:
- Three data fields named width, height and colour that specify the width, height and colour of the rectangle.
 - A constructor to initialise the fields for a Rectangle.
 - A method named getArea that returns the area of this rectangle.
 - A method named isSquare that returns true when the rectangle is a square.
 - A method named draw that draws the rectangle on the UI window at a random place.

```
public class Rectangle{
    private double width;
    private double height;
    private Color colour;

    public Rectangle(double width, double height, Color colour) {
        this.width = width;
        this.height = height;
        this.colour = colour;
    }
    // Methods

    public double getArea(){
        return this.width * this.height;
    }

    public boolean isSquare(){
        return this.width == this.height;
    }

    public void draw() {
        double x = Math.random() * 100;
        double y = Math.random() * 200;
        UI.setColor(this.colour);
        UI.drawRect(x, y, this.width, this.height);
    }
}
```

(b) [5 marks] Complete the following testMethod to create two Rectangle objects, one red with width 100 and height 40, one green with width 10 and height 50. This method should then print the area of the red rectangle and a message showing whether the green one is a square, and draw both objects in the graphics pane.

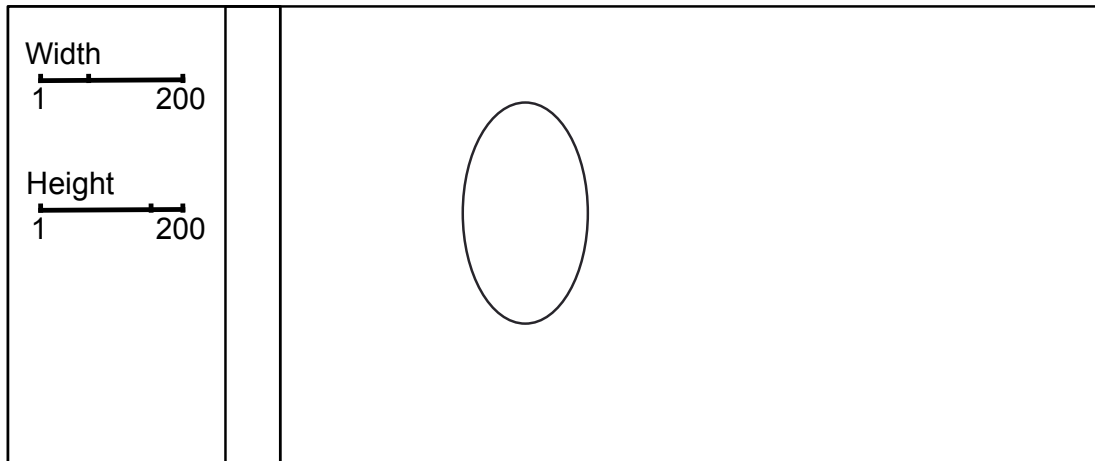
```
public class testRectangle{  
  
    public void testMethod(){  
  
        Rectangle r1 = new Rectangle(100, 40, Color.red);  
        Rectangle r2 = new Rectangle(10, 50, Color.green);  
  
        UI.println (r1.getArea ());  
        UI.println (r2.isSquare ());  
        r1.draw();  
        r2.draw();  
  
    }  
}
```

Question 4. Event-Driven Input**[12 marks]**

Complete the OvalDrawer program on the facing page so that it allows the user to draw ovals with different sizes on the graphics pane.

The program should have two sliders, "Width" and "Height", which allow the user to set the width and height of the oval to be drawn. The maximum size should be 200.

When the user releases the mouse at a position on the window, the program should draw an oval of the specified size centered at that position.



Hint: The addSlider method requires a name, a minimum value, a maximum value, and a method to call.

(Question 4 continued)

```
public class OvalDrawer {

    private double width;
    private double height;

    public OvalDrawer(){
        this.setupGUI();
    }
    public void setupGUI(){
        UI.addMouseListener(this :: doMouse);
        UI.addSlider("Width", 1, 200, this :: newWidth);
        UI.addSlider("Height", 1, 200, this :: newHeight);

    }
    public void newWidth(double w){
        this.width = w;
    }

    public void newHeight(double h){
        this.height = h;
    }

    public void doMouse(String action, double x, double y){

        if (action.equals("released")){
            UI.drawOval(x-this.width/2, y-this.height/2, this.width, this.height);
        }

    }
}
```

Question 5. Files**[12 marks]**

A researcher has done a survey on the number of hours people spent on study, work and volunteering each week. The survey data for each person is saved in one line, including the age, the gender, the ethnic group, and then three numbers on the hours spent on study, work and volunteering.

For example, the file might contain:

```
23 M NZ Māori 20 30 2
29 F European 10.5 30 0
31 F European 24.5 17 0
25 M European 23 20 8.5
```

The first line is about a 23 year old male who is NZ Māori, and spent 20 hours on study, 30 hours on work and 2 hours on volunteering.

Complete the reportTotalHours method on the facing page to read the data from the file, and print out the age, gender, ethnic group and the total hours of each person in the file.

For example, with the file above, reportTotalHours should print out

```
23 M NZ Māori 52.0
29 F European 40.5
31 F European 41.5
25 M European 51.5
```

You may assume that the file is well formatted and there is not any missing data. The only item that is more complicated is the “ethnic group” which may have more than one word, such as “NZ Māori”, “New Zealand Māori”.

(Question 5 continued)

```
public void reportTotalHours(String filename){
    try{

        Scanner scan = new Scanner (new File(filename));

        while (scan.hasNextLine()){
            Ul. print (scan.next()+ " ");
            while (!(scan.hasNextDouble())){
                Ul. print (scan.next()+ " ");
            }
            double total = 0;
            for( int i = 0; i<=2; i++) {
                total = total + scan.nextDouble();
            }
            Ul. println ( total );
        }

    } catch(IOException e){Ul. println ("Fail: " + e);}
}
```
